An empirical study of the "prototype walkthrough": a studio-based activity for HCI education

An Empirical Study of the"Prototype Walkthrough": A Studio-Based Activity for HCI Education

C.D. HUNDHAUSEN,* D. FAIRBROTHER,† M. PETRE††

*Human-centered Environments for Learning and Programming (HELP) Lab, School of Electrical Engineering and Computer Science

†College of Education

Washington State University, Pullman, WA

††Faculty of Mathematics & Computing

The Open University, Milton Keynes, U.K.

For over a century, *studio-based* instruction has served as an effective pedagogical model in architecture and fine arts education. Because of its design orientation, human-computer interaction (HCI) education is an excellent venue for studio-based instruction. In an HCI course, we have been exploring a studio-based learning activity called the *prototype walkthrough*, in which a student project team simulates its evolving user interface prototype while a student audience member acts as a test user. The audience is encouraged to ask questions and provide feedback. We have observed that prototype walkthroughs create excellent conditions for learning about user interface design. In order to better understand the educational value of the activity, we performed a content analysis of a video corpus of 16 prototype walkthroughs held in two HCI courses. We found that the prototype walkthrough discussions were dominated by relevant design issues. Moreover, mirroring the justification behavior of the expert instructor, students justified over 80 percent of their design statements and critiques, with nearly one-quarter of those justifications having a theoretical or empirical basis. Our findings suggest that PWs provide valuable opportunities for students to actively learn HCI design by participating in authentic practice, and provide insight into how such opportunities can be best promoted.

Categories and Subject Descriptors: K.3.2 [Computer and Information Science Education]: *Computer science education, Curriculum*; H.5.2 [User Interfaces]: *Prototyping, User-centered design, Evaluation/methodology*

General Terms: Design, Human Factors, Experimentation

Additional Key Words and Phrases: Studio-based learning and instruction, prototype walkthrough, design crit, HCI, user interface design, video analysis

ACM File Format:

HUNDHAUSEN, FAIRBROTHER, D., AND PETRE, M. (2012). An Empirical Study of the "Prototype Walkthrough": A Studio-Based Learning Activity for HCI Education. *ACM Trans. Computer-Human Interaction.*, x, y, Article z (month year), xx pages. DOI = 10.1145/1290002.1290003 http://doi.acm.org/10.1145/1290002.1290003

---

1. INTRODUCTION

For over a century, studio-based learning (SBL) has served as an effective pedagogical model in architecture and fine arts education. SBL can be conceptualized as an iterative process of solution refinement involving two key learning activities:

- The *design studio* is a shared physical space in which students work on assigned problems. In this space, students are able to see what others are up to, bounce ideas off each other, and help each other.
- In *design crits* (or *critiques*), students present their evolving solutions for feedback and discussion. These take place informally with the instructor, or more formally with the entire class and even industry professionals.

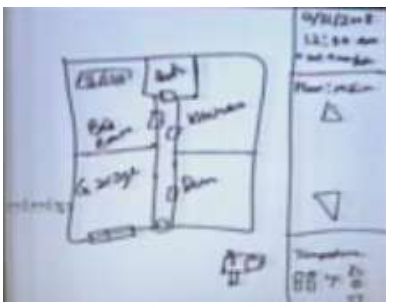
User interface design is a central skill taught in an undergraduate computer science course on human-computer interaction (HCI). In such a course, students often undertake a capstone design project that takes them through all phases of the *user-centered design process* [see, e.g., Preece et al. 2002], including initial data gathering, user interface prototyping, and usability testing. Because of its focus on design, an undergraduate human-computer interaction (HCI) course has been identified as an excellent candidate for studio-based instruction [see, e.g., Arvola and Artman 2008; Reimer and Douglas 2003; Kehoe 2001a].

Within the context of a multi-institutional research project that is adapting and refining the SBL model for computing education [Hundhausen et al. 2008a], we have been exploring the *prototype walkthrough* (PW)—an adaptation of the SBL design crit for an HCI course. In preparation for PWs, student capstone project teams develop low fidelity user interface prototypes of their evolving project designs (see Fig. 1a), and a set of five core tasks to be completed with their prototypes (see Fig. 1b). In sessions lasting approximately 25 minutes each, project teams simulate their low fidelity prototypes on a large projected screen in front of the class. A student from the audience serves as the test user by interacting with the prototype, thinking aloud in the process. At any point, the audience can jump in with questions, comments, or feedback. After the five tasks have been completed, the instructor invites the class to engage in a reflective design discussion intended to help the project team improve its design.

Our exploration of the PW activity raises a pair of basic research questions regarding its educational value as part of an HCI course:

RQ1:	*To what degree does the PW promote discussions that are relevant to user interface design?*

RQ2:	*To what degree do students participate in those discussions?*



(a) Snapshot of low fidelity prototype

Tasks

1. Set thermostat to 88 degrees.
2. Set the attic fan speed to low.
3. Turn on oven and set it to bake at 425 degrees for 40 minutes.
4. Check to see when the last time mail was delivered.
5. Set alarm to wake you up at 1 am on Wednesday.

(b) Five tasks to be completed with the prototype

Fig. 1. Sample Materials Used for Prototype Walkthrough of Team 8's Smart Home Control System

We find *situated learning theory* [Lave and Wenger 1991] to be useful in accounting for the educational benefits of PWs. According to the theory, PWs facilitate learning by providing students with opportunities to participate, in increasingly central ways, in the practices of a disciplinary community: as observers, discussants, test users, and presenters. On this view, learning comes about through *changes in identity* facilitated by such participation, which, because it is mediated by learner-constructed artifacts, is seen as especially valuable in bridging the gap between expert and novice perspectives [Lave 1993].

In the design discussions that take place within PWs, participants' design knowledge manifests itself most readily in the ways in which, and extent to which, design critiques and suggestions are justified. This observation led to an additional research question regarding the educational value of PWs:

RQ3:	*How, and to what degree, are design critiques and suggestions justified?*

Finally, given our interest in better understanding how best to implement PWs so as to maximize opportunities for design learning, we were interested in a fourth research question:

RQ4:	*In what ways does design discussion vary by PW session? What features of individual PW sessions might be linked to those variations?*

This article addresses these questions by presenting a detailed content analysis of a video corpus of 16 PWs that took place within successive offerings of the conjoint undergraduate/graduate human-computer interaction course at Washington State University. In so doing, it makes the following contributions to the HCI education literature:

1. It formulates the PW as a practical studio-based learning activity for HCI education.
2. It presents a rigorous content coding scheme that can be used to analyze critical discussions about user interface design.
3. It provides a rich, descriptive account and theoretically-grounded analysis of the design discussions promoted by PWs, thus providing evidence of their educational value.
4. It provides practical guidelines for instructors interested in incorporating PWs into their own courses.

The remainder of this article is organized as follows. Section 2 presents the background and related work on which our study builds. Section 3 details the design of our study. Section 4 presents the study's key quantitative results, while Section 5 presents a qualitative analysis of the design discussions that took place within PWs. Section 6 discusses our findings vis-à-vis our research questions. Finally, Section 7 considers the implications of the study for computing education research and practice, and identifies directions for future research.

2. BACKGROUND AND RELATED WORK

A form of "design crit" in the SBL model, the PW activity explored in our study engages students in discussions with experts about their user interface designs and how to improve them. A rich legacy of empirical work, nicely synthesized by Cross [2001], has explored the behaviors, activities, and processes of both novice and expert designers. In a similar vein, the large body of research on computer-supported collaborative learning is replete with detailed content analyses of discussions that take place during learning activities, with a focus on how representations serve to mediate those discussions [see, e.g., Roschelle 1996; Suthers and Hundhausen 2003]. The study presented here contributes to all of these lines of work by performing the first detailed content analysis of critical discussions about user interface design within the context of a course on human-computer interaction design.

2.1 Critical Design Dialog

Kehoe [2001] calls the kind of "design crit" on which this study focuses *critical design dialog*, and points out that it differs from other forms of learning discussions in that it is directed toward critiquing students' work in a public forum, with the dual-aim of (a) influencing the trajectory of the work, and (b) providing opportunities for students to learn from each other's design work and feedback. Kehoe [2001; see also Reimer and Douglas 2003] makes a strong case for the educational value of critical design dialog as a means of learning about HCI design. In brief, she argues that the kinds of design problems that are common in HCI are fuzzy and have no clear-cut solutions. Design principles and heuristics that might guide one to solutions are necessarily vague; learners often find them to be unclear and overly ambiguous [Schön 1990], leading to their getting stuck during the design process [Sachs 1999]. Learners, she argues, can therefore best develop design competence when they (a) receive feedback on their own designs that is also connected to more general design principles and heuristics, and (b) observe how experts think about design. Critical design dialog provides ideal conditions for both.

In addition to Kehoe's arguments in favor of critical design dialog, we believe that it is especially appropriate for HCI education because it engages students in the acts of communication, critical thinking, and collaboration. Thus, it can help prepare students for future careers in the software and design professions, which increasingly covet these so-called "soft" skills [see, e.g., Barker 2011].

2.2 Situated Learning Theory

In addition to Kehoe's arguments in favor of critical design dialog as a valuable HCI learning activity, the activity has a strong foundation in *situated learning theory* [Lave and Wenger 1991]. According to this theory, one gains competence within a community of practice by having opportunities to participate, in increasingly central ways, in the practices of the community. Critical design dialog, as manifested in the PW, provides such multi-faceted opportunities for participation. In PWs, students can observe expert critiques of design, remaining on the periphery of the discussions as audience members. As they become more comfortable, they can gradually explore opportunities to offer their own critiques and suggestions. As design team members, students are placed in the

position of presenting, justifying, and defending their own designs. This constitutes more central participation in design practice; it is akin to the situation of an expert designer presenting a user interface design to a design team in a real-world company.

2.3 Wizard of Oz Studies

While we are perhaps the first to use the term "prototype walkthrough" to describe a structured SBL activity for HCI education, we are by no means the first to advocate the activity as a means of identifying usability issues and improving the design of a user interface. In the field of HCI, "Wizard of Oz" studies are a standard means of obtaining early design feedback on low fidelity prototypes [Wilson and Rosenberg 1988]. They have also been used to explore intelligent and futuristic systems [see, e.g., Maulsby et al. 1993]. Prototype walkthroughs can be seen as a kind of "Wizard of Oz" study performed for educational purposes in front of an audience of learners and experts who are invited to participate in the process.

3.4 Studio-Based Learning in Computing and HCI Education

Computing educators have explored the use of SBL in individual computing courses [Myneni et al. 2008] and even in entire degree programs [e.g., Docherty et al. 2001; Tomayko 1996]. In one of the few published studies of SBL in HCI education, Reimer and Douglas [2003] describe their implementation of an undergraduate HCI course based on the studio model. The course included weekly design crits that were similar in spirit to the prototype walkthroughs described here. The key difference was that, in the design crits, the design teams themselves simulated their user interfaces for demonstrational purposes, rather than enlisting a student audience member as a test user. While Reimer and Douglas did not perform a video analysis of their design crits, their observation that the design crits "fostered a highly interactive and constructive learning atmosphere" [Reimer and Douglas 2003, p. 201] resonates well with the findings presented here.

In a similar vein, Cennamo et al. [2011] perform a detailed qualitative comparison of design studios in industrial design and human-computer interaction courses, gleaning insights into how these studios promoted the generation and analysis of design ideas. Likewise, Arvola and Artman [2008] compared HCI students' studio work in a traditional space against that in a computer-augmented space. While their study focused on studio activities that were far less structured than the prototype walkthroughs we studied, it is similar to our study in that it performed extensive analyses of video footage.

3. EMPIRICAL STUDY DESIGN

We conducted our empirical study in conjunction with the spring 2007 and spring 2008 offerings of CptS 443/543 ("Human-Computer Interaction"), the conjoint undergraduate/graduate HCI course at Washington State University taught by the first author. Using a mix of lecture and small group activities and a pair of textbooks [Preece et al. 2002; Norman 2002], the course explored a standard curriculum that focused on the application of relevant theories, principles, and processes to the design of interactive software. A focal point of the course was a capstone user interface design project, which students were required to complete in groups of 2 to 3. Student groups could choose the focus of their projects, or they could take on a project suggested by the instructor. During the 10th week of the 15-week semester, project groups presented prototypes of their

evolving designs to the class within PW sessions scheduled during the regular course lecture periods. These were the focus of this study, which are described in further detail below.

3.1 Participants and Their Projects

The spring 2007 course offering enrolled 13 upper-division undergraduate and 2 graduate students, while the spring 2008 course offering enrolled 13 upper-division undergraduate and 10 graduate students. All but 4 of these students were computer science or computer engineering majors who had minimally completed a sequence of core courses in software design. The other four students came from a mix of majors, including geology and management information systems. None had taken a prior course in HCI.

This study considered the PWs of all 7 project groups in the 2007 course offering, and 9 of the 11 project groups in the 2008 course offering (2 were missed because of technical difficulties with the video equipment). Table I presents the key attributes of the 16 project groups whose PWs were considered in the study.. As can be seen, the projects on which they focused were diverse. Moreover, whereas project groups in the 2007 course offering constructed their prototypes mostly out of simple art supplies (pen, paper, transparencies), most project groups in the 2008 course offering constructed their prototypes using WOZ Pro [Hundhausen et al. 2007], a computer-based low fidelity prototyping tool we developed specifically for this purpose. (See [Hundhausen et al. 2008b] for a detailed empirical comparison of the prototype construction process using art supplies vs. WOZ Pro.)

3.2 Prototype Walkthrough Procedure

Prior to participating in the PW sessions, project teams were required

(a) to perform at least two early data gathering activities (e.g., interviews, questionnaires, field observation) in order to establish the functional, usability, and user experience requirements for their project;
(b) to design a low fidelity user interface prototype based on those requirements; and
(c) to formulate a set of five core tasks that their prototype had to support.

Project teams brought the prototype and set of tasks to the PW sessions, which took place in a small classroom during the two 75-minute lecture periods of the tenth week of the semester.

All students were required to attend and participate in the PWs. Each project group was assigned a 25-minute slot for their PW; students whose group was not immediately presenting were required to observe the PWs, and to fill out a structured evaluation form intended to provide feedback for the presenting project team. Prior to participating in the PWs, students were given the opportunity to sign an informed consent form authorizing them to be videotaped and authorizing the release their data for study purposes; all students chose to sign the form.

Each walkthrough began with the instructor calling a project group to the front of the room. The project team chose a member of the audience to serve as the "test user" for the

Table I. Key attributes of the Project Teams Studied. Teams 1–7 participated in the 2007 course offering, while teams 8–16 participated in the 2008 course offering.

Team	Size	Project Focus	Prototyping Technology Used
1	3	Distributed team problem management	Art supplies
2	2	Smart home event scheduler	Art supplies
3	2	Campus map route finder	HTML
4	2	Personal travel blog site	Art supplies
5	2	N-body simulator	Art supplies
6	2	DVR with remote control	Art supplies
7	2	Power utility mapping software	Art supplies
8	2	Smart home control system	WOZ Pro
9	2	Low fidelity UI prototyping tool	WOZ Pro
10	2	Custom grid-based game builder	HTML
11	2	Poker game	Power Point
12	3	Online code review environment	WOZ Pro
13	2	Campus map route finder	Power Point
14	3	Recipe management software	WOZ Pro
15	2	Custom Monopoly game builder	WOZ Pro
16	2	Group collaboration tool	HTML

walkthrough, subject to the rule that no student could serve as the "test user" more than once. The team provided a brief description of the prototype interface they had designed, along with a general task scenario for the walkthrough. At this point, the project group handed the test user a written set of task instructions, and projected their prototype onto a large screen at the front of the room. Depending upon the prototyping technology used, either an overhead projector, document camera or LCD projector was used for this purpose.

The test user proceeded to complete the set of tasks as the project group simulated their prototype's user interface. The test user was instructed to read each task aloud prior to performing it, and to think aloud while performing each task. In order to perform tasks, the test user pointed directly at and manipulated elements of the image projected on the large screen, describing his or her actions along the way (see Fig. 2). Audience members and the instructor were invited to interrupt the walkthrough at any time with questions or comments. The walkthrough ended when the test user completed all five tasks, all design discussions ended, or the 25-minute time limit had been reached. To conclude each PW session, the instructor initiated a round of applause and called on the next scheduled project group.

In two cases (Sessions 2 and 12), the instructor failed to cut the discussions off at 25 minutes due to his involvement in ongoing discussions. In contrast, because they began too close to the end of the class period, the instructor had to cut off three other sessions (3, 7, and 16) before they had run their complete course.

3.3 Data Collection and Analysis Method

Using a video camera positioned near the middle of the classroom and focused on the projected screen, we obtained 4.91 hours of high-quality video footage of the 16 PWs. Since the research questions posed for this study all relate to the focus and content of the discussions generated by the PW activity, we chose to employ *content analysis* [see, e.g., Krippendorff 1980] as our primary analysis method. We began by partitioning the talk

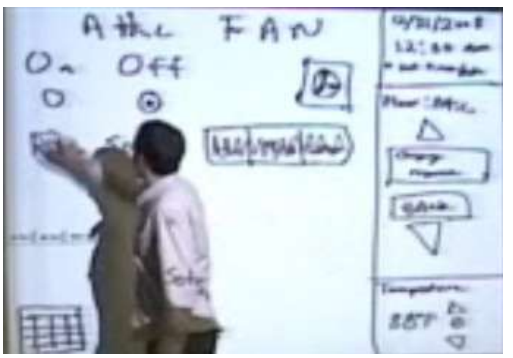


Fig. 2. The Test User Interacting with the Smart Home Control System Prototype in Session 8

into *segments*, where a segment was defined as a single thought or idea uttered by a single participant. We then iteratively developed the coding schemes described below by watching a subset of the walkthrough sessions and adding and refining categories until no new ones emerged. As we did this, we composed a coding manual with detailed categorical descriptions, rich examples of how to distinguish among categories, and step-by-step instructions for coding. Those interested in using or adapting our coding schemes should consult this manual, which is available online [Hundhausen et al. 2009].

Table II presents and briefly describes the nine top-level categories in our content coding scheme. Because of its perceived relevance to the HCI course, DESIGN TALK[1] was of particular interest in this study. Table III presents a more detailed look at DESIGN TALK in terms of its six subcategories. While they are intended to provide an overall feel for the categories, we emphasize that the descriptions provided in these tables are necessarily terse, and lack sufficient detail and examples for one to make reliable distinctions. For more detailed descriptions, we refer interested readers to the coding manual cited earlier [Hundhausen et al. 2009]. The categories in these tables are listed in order of decreasing priority. In cases in which, despite our detailed categorical definitions, we felt a given segment could be coded into multiple categories, we always coded the segment into the category with the highest priority.

In order to gauge the extent to which students and the instructor participated in PW discussions (RQ2), we additionally classified each segment according to role of the participant who uttered it:

- *instructor*—the course instructor and first author of this paper, a computer science professor with two years of industrial experience as a usability engineer and HCI consultant;
- *design team member*—a member of the two or three-person student team whose prototype was being tested;
- *audience member*—a member of the student audience observing the PW;
- *test user*—the student who acted as the test user; and

[1]Throughout this article, categories defined in our coding scheme are written in SMALL CAPS.

Table II. Top-level content coding categories

Category	Description	Examples
DESIGN TALK	Talk focused on design, including justifications, critiques, suggestions, issues, and strategies	See Table III.
USER INTERFACE TALK	Talk focused on the functionality and appearance of the user interface being tested.	"There's a button at the bottom of the screen." "Show me how this works."
TASK DESCRIPTION TALK	Talk focused on the task being performed in the PW	"Was I supposed to do Task 1?"
TASK EXECUTION TALK	Talk focused on what the test user is doing or thinking as she performs tasks	"I'm clicking here, and I want to change the date."
ACTIVITY TALK	Talk directed toward running the PW activity	"It's your turn now." "Any questions?"
PROJECT TALK	Talk focused on the scope or focus of the group's project	"Our project is building a smart home interface."
TOOL TALK	Talk focused on the prototyping technology being used in the PW	"These sticky notes are awkward." "Go to "Run" mode in WOZ Pro."
LAUGH	Laughter uttered by at least two people	[Laughter]
OFF TASK TALK	Talk unrelated to the PW activity	"My job interview went well!"

Table III. Design Talk subcategories

Category	Description	Examples
JUSTIFICATION	Justifies statements about design (critiques and suggestions) or the design of the interface under test	See Table IV.
CRITIQUE	Makes a statement about the goodness of the design	"I don't think that design will work."
SUGGESTION	Suggests an alternative design	"I think you should re-label the button."
ISSUES & STRATEGIES	Discusses design issues, assumptions, strategies for arriving at new designs, and tradeoffs among design alternatives	"You might need to iterate again on this."
META-TALK	Design talk that transcends the specific design under consideration, including comparisons with other designs and general conceptual design issues	"It's a question of how to do good layout." "The issue really is, what is a screen link?"
ENCOURAGEMENT	Congratulates the designers or encourages them to continue their work	"You'll get there." "Keep at it."
RESPONSE	Responds to design talk or expresses understanding of design.	"Good point." "That's a tough call." "I see where you're coming from."

- *class*—at least two speakers in any of the previous speaker categories (reserved only for segments coded as *Laugh*).

Recall that RQ3 focuses on exploring justifications of design critiques and suggestions. To that end, we developed a scheme for classifying design justification statements according to the *basis* of the justification. Table IV describes the twelve design justification categories in this scheme. These categories are listed approximately

Table IV. Design justification categories

Category	Description	Examples
DESIGN PRINCIPLE	Appeals to, or implicitly enlists, an established design principle.	"That's a poor natural mapping." "The user won't know what to do next" (refers to cognitive walkthrough).
TEST USER BEHAVIOR	Based on what the test user actually did, thought, or expected during the prototype walkthrough.	"The user stumbled when he saw it."
PAST USER BEHAVIOR	Based on the behavior of a user in a past user study (carried out prior to the PW).	"Users had trouble with this in a previous study."
OTHER COMPUTER SOFTWARE	Appeals to the design of other similar computer software.	"I think the way Photoshop does it is better."
LIMITATIONS OF PROTOTYPING TECHNOLOGY	Based on perceived limitations of the prototyping software used for the walkthrough.	"Art supplies made it difficult to create polished buttons."
IMPLEMENTATION DIFFICULTY	Based on the perceived difficulty of implementing the design in a given way.	"Ideally, you could freeform draw it, but that would be hard to implement."
LIMITATIONS OF PW ACTIVITY	Based on limitations of the PW activity, including the tasks and their ordering.	"The fact that he did these tasks in a certain order gave him an advantage."
PERSONAL EXPERIENCE	Based on the speaker's personal experience.	"When I've done this in the past, I've always had trouble with making tables."
HYPOTHETICAL USER	Appeals to what a hypothetical user might do or think in given set of circumstances	"A user wouldn't know how to interpret that."
COMMON SENSE	Based on common sense, or day-to-day reasoning.	"The label needs to be changed because the current label doesn't make sense."
NO BASIS	Justification that has no apparent basis	"It's the best we could come up with."

from strongest to weakest, based upon the extent to which they are grounded in theoretical and empirical evidence. The top four categories are rooted in either established design principles (e.g., those described by Norman [2002]) or empirical evidence. Categories that appear further down the table have more to do with personal experience, intuition, or practical concerns. The last category in the table accounts for justifications with no apparent basis.

To address RQ3, we also classified the *strength* of design justifications in terms of (a) the number of unique justifications that were offered on behalf of a given design critique or suggestion, and (b) the strength of the link between the justification and the design critique or suggestion it was meant to justify. Based on our initial analysis of the video corpus, we defined five different levels of strength:

- *Justified directly more than once*—At least two unique justification statements exist that (a) appear in close proximity to the statement they justify (we established that within 10 segments constituted "close proximity"), and (b) are uttered by the same speaker who made the design statement.
- *Justified directly once*—Same as *directly justified more than once,* except that there is only one distinct justification statement.
- *Justified indirectly*—One or more justification statements exist that are (a) uttered by a speaker who is different from the speaker who made the design statement they justify, and/or (b) reside more than 10 segments away from the design statement they justify.

- *Justified implicitly*—No explicit justification statements appear on behalf of a given design statement; however, some previous discussion within the current PW, when taken as a whole, implicitly justifies the statement.
- *Not justified*—A given design statement is not supported by any justification statements.

In order to verify the reliability of our coding schemes, the first and second authors independently coded a 20 percent sample of the video corpus with respect to both the content and the justification strength schemes. We attained a level of agreement of 84 percent (0.82 kappa). Having reached a high level of inter-rater reliability, we had the second author code the remainder of the video corpus.

4. RESULTS

Table V presents summary statistics on our coding of the 16 prototype walkthroughs in our corpus. On average, a prototype walkthrough session lasted 18.4 minutes (SD = 7.4), and contained 176.9 coded segments (SD = 61.1), including 54.2 design talk segments (SD = 36.8) and 16.5 justification segments (SD = 11.6). Session length was strongly correlated with the number of segments in the session (r = 0.687, p = 0.003).

Below, we present a detailed quantitative analysis of our coding of the video corpus, organized around the four research questions posed for this study. Table VI summarizes the key findings. Throughout this section, we treat the *individual PW session* as the unit of analysis.. Except were explicitly noted, the percentages reflect the *mean* percentages of categorized talk across the 16 PW sessions, not the *overall* percentages of categorized talk in the 16 PW sessions combined. Analyzing the data in this way gives equal weight to each PW session, rather than weighting each session by its length. We believe that this is an appropriate way to analyze the data because it reflects the reality that each PW session was largely independent of the others. Indeed, while following an identical protocol, each PW session had a different user interface design as its focus, and different students in the three student roles (design team, audience members, and test user).

4.1 PW Discussions: Content and Contributions

We first explore our data relevant to RQ1 ("To what degree does the PW promote discussions that are relevant to user interface design?") and RQ2 ("To what degree do students participate in those discussions?"). Fig. 3 presents the mean percentage of talk dedicated to each of the high-level content categories in our video corpus within a PW session. Within each category, the talk is broken down further by participant type.
As Fig. 3 indicates, three categories of talk dominated the PW discussions:

- DESIGN TALK (M = 27.8%, SD = 13.5%), which focused on actual user interface design issues;
  USER INTERFACE TALK (M = 24.7%, SD = 11.6%), which focused on helping partici pants better understand the user interface being tested; and
- TASK EXECUTION TALK (M = 23.0%, SD = 10.1%)—the test user's think aloud protocol, which provided a basis for evaluating the strengths and weaknesses of the user interface being evaluated.

Table V. Summary Statistics on PW sessions

Session	Duration (Minutes)	# Segments	# Design Talk Segments	# Justification Segments
1	21.5	143	58	12
2	24.0	160	34	5
3	8.1	100	3	0
4	15.6	101	32	12
5	34.1	216	85	39
6	24.1	195	77	36
7	13.4	126	7	4
8	8.8	100	24	13
9	18.0	241	102	34
10	19.1	190	44	20
11	17.5	236	108	23
12	27.5	259	75	23
13	21.5	238	65	30
14	22.4	270	116	32
15	8.5	108	18	7
16	11.0	148	19	8

Table VI. Summary of Key Findings of Quantitative Analysis

Finding	Relevant RQ	See Section
1. PWs focused on issues relevant to an HCI Course, with DESIGN TALK and USER INTERFACE TALK consuming a majority (53%) of the discussions.	1	4.1
2. Contributions to overall talk differed significantly by participant role, with the Instructor and Design Team contributing significantly more DESIGN TALK and JUSTIFICATION segments than the Test User and Audience	2	4.2
3. 60% of participants' design statements were directly supported with one or more justifications. 30% of those justifications had an empirical or theoretical basis. No statistically significant differences could be detected between students and the instructor with respect to either of these measures.	3	4.2, 4.3
4. The quantity of Design Talk and Justification segments varied widely by PW session, and was strongly correlated with the length of the session, but unrelated to when the PW session occurred in time or the grades of the design teams who presented in the PW session.	4	4.4

Inspection of Fig. 3 suggests that each participant type contributed in different quantities to the PW discussions. Fig. 4 brings this into sharper focus by presenting the mean percent contribution of each participant type. As Fig. 4 illustrates, members of the design team who were simulating their interfaces contributed roughly one-third of the discussion content—the most of any speaker type. Not far behind were the test user, who thought aloud while completing tasks with the design team's prototype interface, and the course instructor, who facilitated the PW sessions; both contributed roughly one-quarter of the discussion content on average. Audience members were not as extensively involved, contributing 10 percent of the talk. The "class" speaker type, used only in conjunction with Laugh segments, contributed just under three percent, reflecting

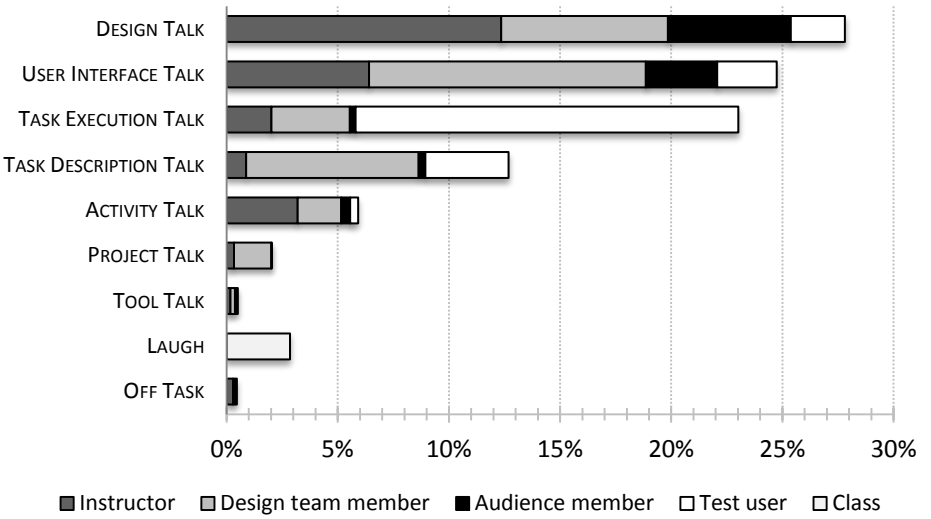


Fig. 3. Mean PW session content classified by top-level content category (see Table II) and participant type. Note that the "Class" participant type appears only in the "Laugh" content.

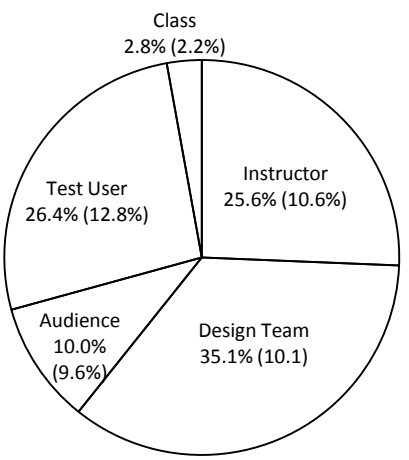


Fig. 4. Mean Contribution of Each Participant Type to all Talk (Standard Deviations in Parentheses)

fact that, on average, roughly three percent of PW discussion content consisted of laughter.[2]

In looking at Fig. 3, one also sees that each participant type contributed different types of talk to the PW sessions. According to a chi-squared test of homogeneity, the distribution of talk across our high-level content categories varied significantly by

[2] Subsequent analyses ignore the "class" speaker type, which, by definition, contributed only Laugh segments.

participant type, $\chi^2(18, N = 2741) = 1370.0$, $p = < 0.0001$.[3] This result reflects the fact that each participant type played a different role in the PW sessions. Test users were involved in performing tasks, and hence contributed more TASK EXECUTION TALK and TASK DESCRIPTION TALK than the other participant types on average. Because they were responsible both for describing the tasks to be performed and for simulating their interface for those tasks, design team members contributed more TASK DESCRIPTION TALK and USER INTERFACE TALK than the others; they also contributed the most PROJECT TALK, which focused on their overall interface design projects, including its background and history. Finally, the instructor contributed the most DESIGN TALK, perhaps owing to his position as the course instructor and expert user interface designer.

Because we regarded DESIGN TALK and JUSTIFICATION (a subcategory of DESIGN TALK) segments both as most relevant to the course, and as most indicative of (emerging) design expertise, we were interested in testing for differences in the percentages of DESIGN TALK and JUSTIFICATION segments contributed by each participant type. According to ANOVA,[4] there did indeed exist a statistically reliable differences between the percentage of DESIGN TALK segments, $F(3, 60) = 9.22$, $p < 0.0001$, $\eta_p^2 = .316$, and JUSTIFICATION segments, $F(3, 60) = 5.52$, $p = 0.002$, $\eta_p^2 = .216$. Both tests had high power: 0.995 and 0.926. In both cases, post-hoc Tukey contrasts revealed that the differences ($p < 0.05$) lay between the instructor and test user, and between the instructor and the audience.[5] Notably, no significant difference could be detected between the percentages of DESIGN TALK or JUSTIFICATION segments contributed by the instructor and design team members.

Fig. 5 takes a closer look at DESIGN TALK, breaking it down both by the subcategories described in Table III, and by participant type. As can be seen, roughly one-third of DESIGN TALK statements consisted of CRITIQUES of the user interfaces being presented in the PWs ($M = 12.7\%$, $SD = 23.8\%$), or SUGGESTIONS for improvement ($M = 20.5\%$, $SD = 9.7\%$). Roughly another third of DESIGN TALK statements either justified those CRITIQUES and SUGGESTIONS, or justified the design of the user interfaces being considered in the PWs ($M = 35.1\%$, $SD = 15.3\%$). The remaining third of DESIGN TALK was dominated by discussion of ISSUES AND STRATEGIES, and direct RESPONSES to other DESIGN TALK.

Fig. 5 suggests that participant types contributed in different quantities to DESIGN TALK. Fig. 6 illuminates these differences by presenting the mean contribution of each participant type. As Fig. 6 shows, over 40 percent of DESIGN TALK segments came from the instructor, with design team members contributing roughly one quarter of the segments, and test users and audience members each contributing less than one-fifth of

[3] Chi-squared tests of homogeneity test categorical frequencies; they cannot be applied to session means. However, when we performed chi-squared tests on each of the 16 PW sessions individually, we obtained similar statistically significant results.

[4] We ran all statistical tests using both parametric and non-parametric (Kruskal-Wallis) ANOVAs. In all but one case (see next footnote), the results were identical. In order to provide effect sizes, confidence intervals, and power estimates, we report the results of parametric ANOVAs.

[5] We also detected differences between the design team and test user that approached significance with respect to % DESIGN TALK ($p = 0.052$) and % JUSTIFICATION ($p = 0.068$). Notably, according to non-parametric Bonferoni post-hoc contrasts, these differences were significant in both cases (% DESIGN TALK: $p = 0.006$; % JUSTIFICATION: $p = 0.048$).

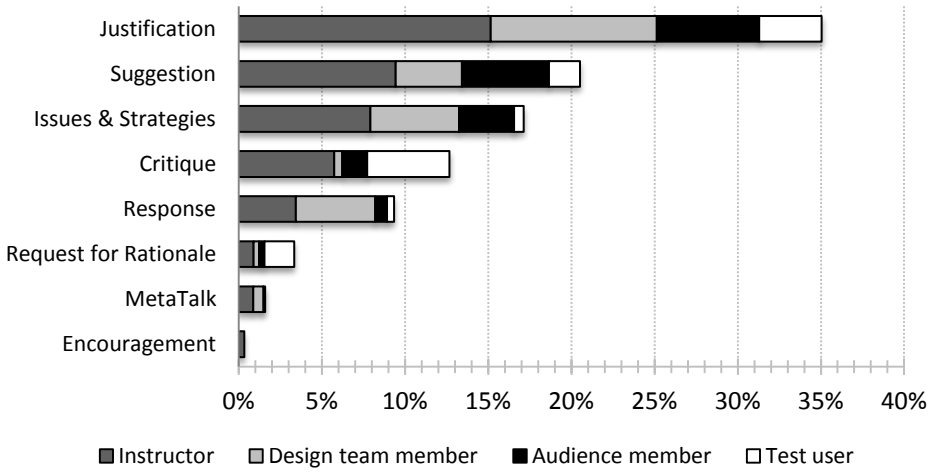


Fig. 5. Design Talk Content by Design Talk Subcategory and Participant Type. Note that Design Talk composed 27.8% (SD = 13.5%) of PW session talk on average.

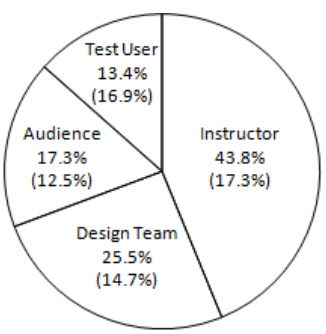


Fig. 6. Mean contribution of participant types to Design Talk (standard deviations in parentheses)

the segments. Interestingly, the four participant types' contributions differ from their contributions to overall talk: Whereas the instructor and audience members contributed a *greater* percentage to DESIGN TALK than to overall talk, design team members and the test user contributed a *smaller* percentage.

Fig. 5 also indicates that participant types contributed different types of DESIGN TALK. A chi-squared test of homogeneity confirms that the distribution of segments across DESIGN TALK subcategories differed significantly by participant type, $\chi^2(12, N = 871) = 68.3$, $p < 0.0001$. This is consistent with the findings for overall talk, and reflects the differing roles that participants played in the PW activity.

4.2 Basis of Justifications of Design Critiques and Suggestions

In this subsection and the next, we shift to an exploration of data relevant to RQ3 ("How, and to what degree, are design critiques and suggestions justified?"). On average, 9.7% (SD = 5.2%) of the segments of each PW session were coded into the

JUSTIFICATION category. Fig. 7 breaks these segments down according to the taxonomy of justification bases presented in Table IV, For each justification basis, a stacked bar additionally indicates the contribution of each participant type.

As the chart indicates, an average of 30.0% (SD = 18.6%) of justifications had an empirical or theoretical basis. In particular, 17.2% (SD = 16.5%) were rooted in empirical evidence (TEST USER BEHAVIOR, PAST USER BEHAVIOR), while another 12.7% (SD = 10.6%) were grounded in established HCI theories, concepts, or principles. Fig. 8 takes a closer look at the 37 justification statements across the entire corpus that were rooted in HCI principles, heuristics, and concepts. The 80-20 rule [see, e.g., Lidwell et al. 2010] and the conceptual model [Norman 2002] were cited most frequently. Other concepts, principles, and heuristics were cited less often: the cognitive walkthrough [Polson et al. 1992], empirically-established limits of cognition and memory (e.g., 7 +/- 2 items can be stored in working memory [Miller 1956]), Nielsen's [1992] heuristics (consistency and standards, minimalist design), Norman's [2002] design concepts (visibility, feedback, constraints), the concept of software scaffolding [Guzdial 1994], the Principle of Direct Manipulation [Shneiderman 1983], and the concept of a transfer effect [see, e.g., Card et al. 1983]. Not surprisingly, all of these had been studied previously in the course.

Of the remaining justifications, appeals to COMMON SENSE (M = 22.9%, SD = 12.5%), a HYPOTHETICAL USER (M = 20.6% SD = 17.2%), OTHER SOFTWARE (M = 8.0%, SD = 7.4%), and PERSONAL EXPERIENCE (M = 5.2%, SD = 7.0%) were most common. Practical concerns, including the perceived IMPLEMENTATION DIFFICULTY of a given design (M = 3.59%, SD = 4.1%), LIMITATIONS OF THE PROTOTYPING TECHNOLOGY (M = 3.3%, SD = 5.5%), and LIMITATIONS OF THE PW ACTIVITY itself (M = 2.9%, SD = 5.9%), were less common. Just 3.6% (SD = 4.7%) of justifications had NO BASIS whatsoever.

We consider empirical evidence and design principles to form the strongest basis for critiques and suggestions regarding user interface design. These are the “good” kinds of

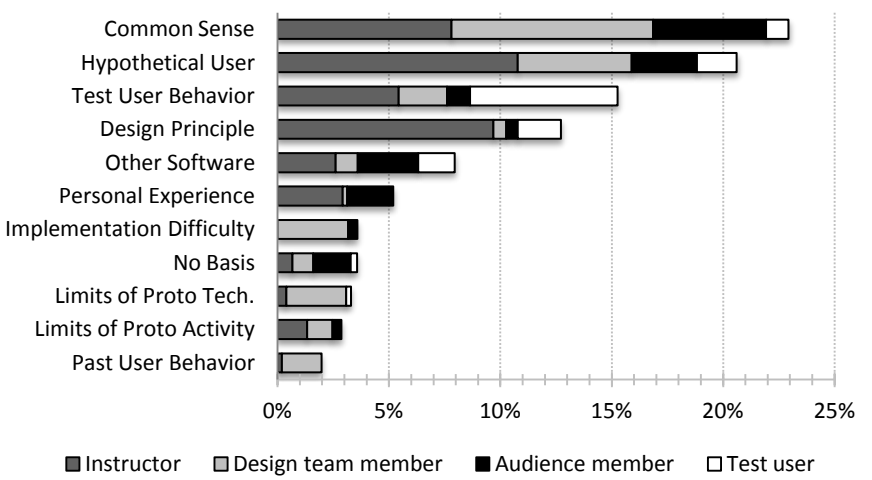


Fig. 7. Justifications classified by basis (see Table 4) and participant type

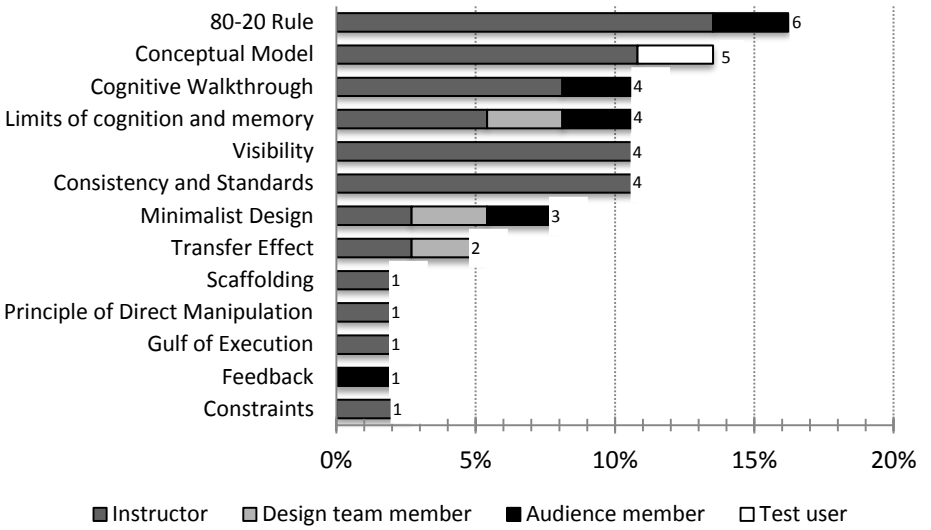


Fig. 8. Justifications based on design principles by speaker type

justifications that HCI instructors would like to model, and that HCI students would ideally learn to enlist within an HCI course. Given this, we wondered whether the instructor (an HCI researcher with two years of industrial experience) enlisted significantly more "good" justifications than the students. To explore this, we pooled (a) DESIGN PRINCIPLE, TEST USER BEHAVIOR, and PAST USER BEHAVIOR into one category ("good" justifications), and (b) the audience, test user, and design team into one category ("all students"). After partitioning our data in this way, we found that, on average, 32.5% (SD = 30.0%, 95% CI [17.4%,47.5%]) of the instructor's justifications, and 24.7% (SD = 26.8%, 95% CI [9.7%,39.8%]) of all students' justifications were "good." According to an ANOVA, the difference was not statistically significant, $F(1,28) = 0.56$, $p = 0.462$, $\eta_p^2 = .020$. However, the estimated power of this test was low (0.111), suggesting that we would need more data in order to be able to detect whether there exists a reliable difference with respect to the goodness of students' and the instructor's justifications.

4.3 Strength of Justifications of Design Critiques and Suggestions

In order to investigate the degree to which participants justified the design critiques and suggestions that they made, Fig. 9 presents the mean percentage of DESIGN CRITIQUE and DESIGN SUGGESTION segments that were justified according to each of the strength levels we defined at the end of Section 3.3. As Fig. 9 indicates, nearly 60 percent of CRITIQUES and SUGGESTIONS were directly justified with either one (M = 46.1%, SD = 25.4%) or more than one (M = 12.8%, SD = 15.9%) JUSTIFICATION statement. Approximately one quarter of CRITIQUES and SUGGESTIONS were justified either indirectly (M = 19.3%, SD = 18.1%) or implicitly (M = 5.5%, SD = 9.1%). Only 16.4 percent of CRITIQUES and SUGGESTIONS (SD = 17.0%) were not justified at all.

Just as we consider that justifications grounded in empirical evidence and established HCI design principles provide the strongest basis for design suggestions and critiques, so

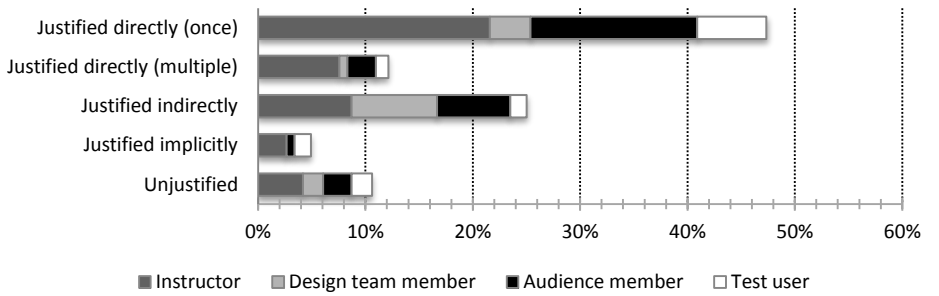


Fig. 9. Justification of design critiques and suggestions by strength and speaker type

too do we consider that multiple, direct justifications provide the strongest basis for critiques and suggestions. Analogous to the analysis of Section 4.2, we wondered whether the course instructor provided significantly stronger justifications than the students. To explore this, we first weighted each justification by its strength. Unjustified statements received 0 points; implicitly justified statements received 1 point; indirectly justified statements received 2 points; directly justified statements received 3 points; and statements with two or more direct justifications received 4 points. After repartitioning participants into two categories (students and the instructor), we found that, on average, the instructor's mean justification strength was 2.61 (SD = 0.83, 95% CI [2.21,3.01]), whereas students' mean justification strength was 2.17 (SD = 0.73, 95% CI [1.77,2.57]). According to an ANOVA, the difference was not statistically significant $F(1,30) = 2.45$, $p = 0.128$, $\eta_p^2 = .075$. However, as was the case for the statistical test of justification goodness reported above, the estimated power for this test was low (0.328). This suggests that we would need more data in order to be able to detect whether there exists a reliable difference between the strength of students' and the instructor's justifications.

4.4 How Design Talk Varied by PW Session

We now analyze the ways in which design discussions varied by PW session, in an attempt to identify features of individual PW sessions that might have led to educationally-relevant discussions about design (RQ4). Fig. 10 and Fig. 11 present the percentage of Design Talk and Justification segments by speaker and session, with the sessions appearing chronologically from left to right. (Recall that the first 7 PW sessions took place in the Spring 2007 course, while the final 9 PW sessions took place in the Spring 2008 course.) Inspection of these figures reveals that the proportion of DESIGN TALK and JUSTIFICATION segments varied widely by PW session; however, the variance appears to be unrelated to when PW sessions occurred in time.

In fact, upon further analysis, we discovered that the presence of Design Talk and Justification segments had more to do with the *duration* of the PW session: Session duration was strongly correlated with both the percentage of design talk segments in the session ($r = 0.61$, $p = 0.01$) and the percentage of justification statements in the session ($r = 0.53$, $p = 0.03$). This implies that when sessions ran longer, they contained greater proportions of Design Talk and justification segments.

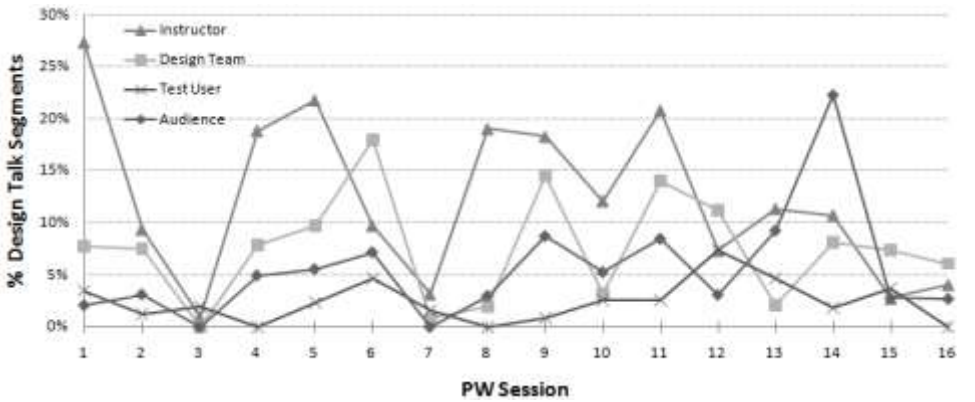


Fig. 10. Percentage of Design Talk Segments by Session and Speaker Type

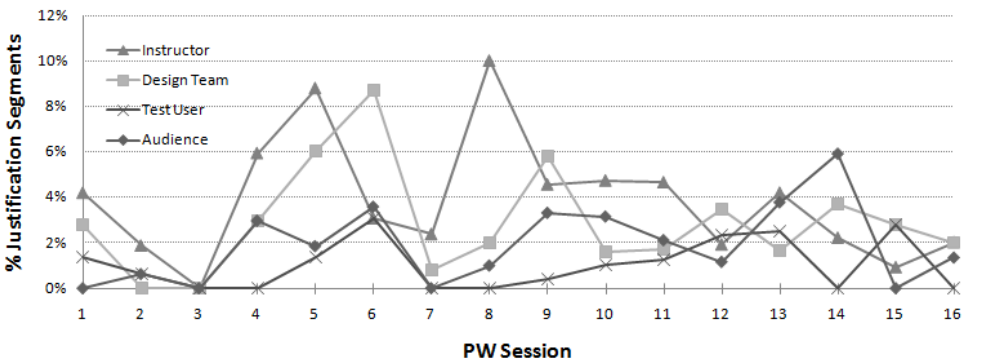


Fig. 11. Percentage of Design Talk Segments by Session and Speaker Type

On the conjecture that stronger design teams might promote higher percentages of Design Talk and Justification segments, we also wondered whether those measures might be correlated with the grades of design teams. However, this was not the case: Grades for the design document that teams submitted a week after the PW sessions were weakly correlated to the percentage of design talk segments ($r = 0.20$, $p = 0.47$) and justification segments ($r = 0.46$, $p = 0.07$) in the corresponding PW session.

5. QUALITATIVE ANALYSIS OF DESIGN DISCUSSIONS

In order to gain further insight into RQ4, we now present a qualitative analysis of the design discussions that took place in the PW sessions. As a starting point for this analysis, we observed that 4 of the 16 sessions (1, 9, 11, and 14) contained over 40 percent DESIGN TALK, whereas another 4 sessions (3, 7, 15, and 16) contained less than 20 percent DESIGN TALK (see Fig. 9). Given our interest in session features that might be linked to variability in DESIGN TALK, we thought that these sessions, which were widely mixed in terms of their percentages of DESIGN TALK, would form an appropriate sample. In qualitatively analyzing these sessions, our objective was two-fold: (a) to identify features of the sessions that might have led to greater or smaller percentages of DESIGN TALK; and (b) to identify themes in the topics and focus of the design discussions. Below,

we present the results of these two analyses, using transcribed vignettes to illustrate and elaborate on our findings.

5.1 Session Features that Influenced Design Talk

We identified four general features of PW sessions that influenced participation and the amount of design discussion, as discussed below.

5.1.1 Application Domain. We observed that sessions that focused on familiar applications tended to stimulate increased participation and design discussion. For example, Session 14 focused on a recipe management application. This idea was met with great interest by participants, leading to the highest rate of audience participation of any session (see Fig. 10). Participants seemed to take great pleasure in brainstorming possible features for such an application, as illustrated in the following exchange:

A1:[6] You could offer, like, a random dinner, select random dinner.

A2: Or "feed me!"

A3: That's pretty good!

A4: That's really good!

[Laughter]

I: That's really good. You gotta put him on the focus group!

A5: That's actually really good, because we end up cooking the same things over and over again. I'd love to have a random dinner.

I: I'd like the random dinner, and the cook for it!

[Laughter]

Likewise, Session 7's focus on the design of a familiar device—a DVR and remote control—captured the attention of the class. One design aspect of particular interest was the layout and color scheme of the remote control's buttons. For example, in the following vignette, an audience member with partial color blindness raises a concern about the color scheme:

A1: With those buttons (refers to currently-displayed screen),…I'm partially color blind, so I'd always have a hard time with that. My thing is like, there's two buttons there—those could be the bottom two or the top two.

A2: That's a good point.

---

[6] In the vignettes presented in this section, we use the following labels to designate speakers: A = audience member, I = Instructor, DT = design team member, and TU = test user. If necessary, we use a secondary numeric label to indicate distinct audience members (e.g., A1, A2) or design team members (e.g., DT1, DT2) in a given exchange.

DT: We considered that. We actually implemented that and then took it out. Yeah, we'll think about that more then.

Reflecting on his experience with interacting with the remote, the test user echoes his concern over the use of color to convey which buttons are currently available:

TU: I was thinking the same thing cuz like, when you had just two [buttons available] I was like, "So there's a blank area above. Does that mean that blank area represents the first button, and the second button is the one I actually want to press?"

This leads to a realization of a design team member, who admits that the team had trouble figuring out how to convey the functionality of the buttons:

DT: Yeah, you have to rely on the color if [the buttons aren't labeled with numbers]…When we looked at it, we were kind of confused. We were like, "There's two buttons there that don't do anything.

The discussion culminates with the test user and audience member collaborating on an alternative design that uses shape, instead of color, to convey whether a button is available:

TU: Maybe it's just like the buttons that are active are square, and then there's just like a little oval, or something like that, that's the same color so that you realize, "Okay, this doesn't do anything."

A1: Or just like a square with no color to let you know there's a button there but don't press it.

A2: Good idea.

In contrast, applications that were obscure or unfamiliar to participants tended to dampen design discussions. Team 7, for instance, designed an application to be used by a local power company to map power lines. In this session, much of the conversation was dedicated to clarifying the obscure application domain and specialized test scenario, as illustrated in the following interaction between a design team member and the test user:

DT: In the current set-up, which is straight from [what they currently use], an object is like the pole, and the wires that go between the poles.

TU: I'm attempting to add a point of some kind, right?

DT: At your current location (points to "map" area of screen).

TU: What happens if I click inside the map somewhere (gestures in the "map" area of screen)?

DT: Then it places an object there. That would be, in this case, a 12-foot tall cedar pole.

A few seconds later, the test user remains confused about the purpose of the task, and asks for re-clarification. Unfortunately, the design team member's explanation is of little help, as illustrated by the test user's resignation and sarcasm in the following exchange:

TU: That is what I want?

DT: Maybe. If that is where you are, but if it wasn't, then it wouldn't be.

TU: If that was where I am. It is, but if it wasn't, it wouldn't be? Okay, sweet, I'll run with that.

Because of struggles like these to gain an understanding of the application domain and tasks, little time remained in this session to discuss the design of the application itself.

5.1.2 Task and Interface Clarity. We observed that the extent to which a design team presented their usage scenario, tasks, and interface in clear and understandable terms influenced the discussions that ensued. If a design team *underspecified* the tasks or interface, participants tended to ask questions that shifted the discussions away from design. For example, Team 15 developed a team collaboration tool. The second task in the PW was to "create a blog post." Some 45 seconds after searching for functionality to perform the task, the test user finally reached a page entitled "Editing a blog." Visibly confused, the test user did not appear sure he was in the right place, whereupon one of the design team members initiates the following exchange:

DT: You're editing a blog, right?

TU: Okay, so.

DT: Have you used blogs?

TU: No.

DT: Okay.

TU: So, this is the new blog, or?

At this point, an audience member and the instructor jump in, for they, too, are confused about the task:

A: This is somebody's—this is already some entry.

I: Did it say "create a *new* blog?"

DT: You wanna create a new blog post.

I: But not a new *blog*?

Nearly two minutes after he began the task, the test user finally arrives at the correct screen for the task, entitled "New Blog." While the design team's user interface certainly contributed to this struggle, this vignette illustrates that the vagaries of the task itself may have been the main culprit in diverting attention away from design.

In a similar vein, we observed that the *fidelity* of design teams' user interface prototypes influenced design discussions. Resonant with prior research into the impact of prototype fidelity on design discussions [Schumann et al. 1996; Hewson 1994], we observed that higher fidelity prototypes tended to discourage discussions about design, because the audience perceived them more as finished products. For instance, Team 3 implemented a campus map interface as a high fidelity prototype on top of Google Maps. As the test user completed a task in which he mapped out a route to the parking lot closest to a given building, he decided to see if he could find another nearby location he was familiar with. This exploration led to the following exchange:

I: Wow!

TU: That was cool!

I: What's the task again? You're trying to get directions?

DT: Yeah.

In the next task, the test user was asked to get driving directions from his house to the place he located in the previous task. Once he had done so, a round of adoration ensued:

TU: Aw, nice!

A: Cool!

TU: There it is.

DT: This is Google Maps, so it's not really us.

Even though the design team had augmented an existing interface with new functionality that had potential problems, the fact that the interface actually worked like Google Maps diverted attention away from the design team's contribution to the design.

On the flip side, we also observed that *underspecified* prototypes could get in the way of design discussions. For instance, Team 14 implemented their custom Monopoly game builder as a *horizontal* prototype that supported superficial tasks like starting a game and making the first move, but did not support the functionality that would be needed to support more complex game scenarios. In the following vignette, the test user is interested in performing functionality that the prototype does not support:

TU: So I'd like to trade. *(Clicks the "trade" button in prototype interface.)*

DT: So, uh.

[Laughter]

I: Now we're going off the map.

DT: So yeah, we gotta end your turn here. Nobody else has any properties. You can only trade for cash.

This prompts the instructor to ask about the unspecified functionality:

I: So "trade" should be grayed out here, huh?

DT: Well, you could trade for cash, but I don't know if anyone would want to buy it at the beginning of the game.

I: How would he do that?

DT: Well, we don't have that [implemented].

…

I: Things that aren't available right now should not be available.

Notice that this sequence culminated with a superficial design suggestion, which was prompted by the lack of functionality specified in the design team's prototype interface.

5.1.3 Problems encountered by test users. User interface designers have long observed that good design goes largely unnoticed, whereas bad design tends to get people's attention. It follows that episodes in which test users struggled to complete tasks provided the best stimulus for grounded discussions about user interface design. Indeed, when observing such episodes, PW participants were presented with opportunities not

only to diagnose what happened, but also to apply the principles, concepts, and theories of the course, and to identify design alternatives to remedy the problem.

In those four sessions in which design talk consumed more than 40% of the overall discussions, test users' struggles were a key catalyst for design discussions. Especially if the problem was relatively minor, those discussions might take place right when the problem occurred. For example, in Session 11, which focused on a Texas Hold 'em poker game, the instructor offers the following design critique immediately after the test user loses a hand:

DT: You lost your money!

I: The problem is, we don't know how much money he lost. You didn't represent it [in the interface].

Alternatively, to stimulate deeper design discussions, the instructor routinely revisited a problem after the test user had completed all tasks. The general strategy was three-fold: (a) recap the interaction, (b) interpret it against the backdrop of particular principles, concepts, or theories being explored in the course; and (c) open up a discussion about how the problem might be remedied. For instance, in Team 5's PW of an *n*-body simulator, the test user could not figure out how to set the properties of a simulation. After the test user had completed all tasks, the instructor raised this issue for discussion:

I: One of my concerns is having simulation properties as a scene item. We saw in the walkthrough that wasn't where the user seemed to look to set the properties. Would you consider locating simulation properties somewhere else? Because if you consider the cognitive walkthrough, the step where that failed was the second step: mapping what to do onto the user interface. They couldn't find that. So I think that might be misplaced.

Having established the problem, the instructor then looked to the interaction itself for guidance on how to remedy the problem:

I: Where did the test user look is the question?

DT: He looked in "Edit" first.

A spirited discussion about where to locate the simulation properties within the interface ensued. Several participants contributed ideas, including this final exchange between two audience members, the design team, and the instructor:

A1: Or just have a big button on right hand side at the top there that says "simulation properties" or something, because., like, I wouldn't think to look in the list box for that stuff. Some sort of like, big button or something that draws your attention to it is important.

A2: But if you also still want to support being able to have stuff in the properties box, you could do both: you could have it as an item, but also have it somewhere in the menu.

DT: It would be tough to draw users' attention to the properties box. It would basically entail making two interfaces: one to go in the properties box, and another to pop up as a dialog box.

I: Yeah, you're fighting that. Should it be dealt with in the same way scene items are dealt with? Or should it have its own special kind of prominence, because it's this global set of properties. . .I guess the question is, do people need them visible all the time, or only when they are setting them?

Thus, what started out in this PW as a simple struggle with locating simulation properties ended up generating a rich discussion about *visibility* and *conceptual model*, two key design concepts explored in the course. This conversation got to the heart of the design choices faced by Team 5, who were drawn to the aesthetics of what they had done, but were struggling to find a design that would be usable to others.

5.2 Design Talk Themes

The preceding example illustrates the power of test user interactions in spurring rich design discussions that consider broader design principles and concepts. In the four "productive" sessions with at least 40% design talk, we identified 22 design discussions, each of which was distinguished by its focus on a particular topic. Further analysis of these discussions revealed that they all focused on one of three distinct themes. Below, we explore each of these themes, in decreasing order of the frequency of their occurrence.

5.2.1 How to make functionality more accessible? 13 of the 22 design discussions focused on how to make a given piece of functionality more accessible to the user. These discussions were invariably generated by the test user's struggles, which led participants to consider *why* the test user struggled, and *how* to redesign the interface to remedy the problem. An excellent example of this kind of discussion was previously presented in Section 5.1.3. Another example of this type of discussion could be found in Session 1, which considered a task and issue tracking tool for collaborative teams. In one task, the test user was asked to search for an issue with a certain title. After typing the problem name into a "problem title" box, the test user received a list of results below that box. A single pane below the results list indicated all of the attributes of the currently-selected search result. After completing the tasks, the test user immediately voiced his concern over this interface:

> TU: I thought it was a little confusing…You had all the attributes visible, and you had a list [of search results] at the same time, whereas, with something like Google, it just lists the title and maybe a little snippet of each [search result], without presenting [the details of a search result] until you actually click it.

The discussion culminated with a brainstorm on how the interface might be redesigned to make the searching functionality more accessible:

> I: In a lot of applications, you would see [the window displaying the details of each search result] in the search pane itself (*points to search pane in display*), indicating those are the things that came up. There could be a usability issue here with not putting the results in spatial proximity to where you're actually seeking the results. . .If nothing else, that label (*points to "Problem title" label on display*) I don't think is suggestive of the fact that they are actually search. . .results.
>
> DT: You could put "attributes" underneath the [search results] list, to let you know that's different than the actual list.
>
> A1: You could have that list box (*referring to search results pane*) take up the entire page, and [only display the item when] they select an item from that box.
>
> A2: What I would suggest is have a results tab. When you click "apply search" it would automatically pop to that tab with the results of the search, and when you select something in that tab it would automatically populate all those fields.

5.2.2 What functionality is most important? A key design principle explored in the course was the so-called "80-20" rule: 20% of an interface's functionality is used 80% of the time [see, e.g., Lidwell et al. 2010]. This principle implies that the most commonly-used functionality should be the easiest to access. In coming to grips with their design domains, teams had to answer a central question related to this principle: What functionality should be featured most prominently in the interface? 5 of the 22 design discussions had to do with this question. For example, near the end of Session 1, the instructor initiated a discussion regarding the overall importance of email functionality in Team 1's collaborative task and issue tracking tool, whose top-level menu consisted of 3 items—"Report Problem," "View/Edit Problem Reports," and "Exit":

I: A priori, looking at these options, and trying to map "E-mail" problem to one of these, I wouldn't be able to do it. . .I guess it depends on how important e-mail is. If e-mail is kind of a minor thing, then I think you could get away with this hierarchy. But if you see it as more of a major task, I'm not sure if this top-level menu is appropriate. . .You could maybe change the label of "View/Edit".

DT: To "Manage"?

I: "Manage" or something, yeah.

A second example of this kind of discussion theme occurred in Team 14's PW, which considered a software tool for recipe management. Perhaps because this domain was so familiar to participants, they identified additional features for the tool not included in Team 14's prototype:

A1: Often, I have a bunch of crap in my refrigerator, and it's like, I need to create something that's edible.

DT: Yeah, we thought of this too, but this tool already has so much in it. There are so many possibilities. We couldn't cover everything!

A1: Yeah, it's too tough.

A short while later, another audience member had an idea:

A1: Do you have any [recipe] filtering based on calories?

DT: Yeah, we have [included that in] the nutritional analysis.

A1: Well, I saw that there is a filter based on time. . . I am interested in a filter based on calories.

DT: Okay, yeah. We just couldn't include everything!

5.2.3 How to support learning. Team 11 designed a tool to enable people to play Texas Hold 'Em Poker. While audience members were familiar with poker, it turned out that few knew about this particular style of poker. An overarching theme of this session, manifested in four different design discussions, had to do with how to design an integrated help facility to support the learning of the game.

These discussions started out by considering how to convey "the flop," which is the game's lingo for the three cards that are displayed after the initial two-card hand is dealt. The instructor begins by raising his concern that "the flop" will be unfamiliar to new players:

I: "See the flop first" *(reading from currently displayed screen).* How does a beginner know what the flop is?

DT1: Ew, good point.

I: I don't know what "the flop" is. I mean, I watch ESPN 2, but. . .I never know what they're doing.

A member of the design team responds with an initial suggestion:

DT1: Would it be more helpful to say, maybe, "the next three cards?"

I: Yeah.

An audience member then counters with another idea:

A1: What I think would be better is something that explained what the game was, if it's a beginner [playing].

A2: Like a manual.

A1: Yeah, like "this is what this means," this is why you have two cards.

TU: That'd be really cool, because otherwise it's like, "What is a flop?"

This exchange leads into another discussion about how to support learning game strategies:

I: Also, I think the other key is, What wins? Like explain. . .

DT1: Does that kinda time with the help? Like the help [will] explain, like, hands and ranks.

Aware of the importance of helping people learn within the context of tasks, as opposed to outside of them, the instructor nudges the design team toward a scaffolded learning approach:

I: In context, though. For example, you have a suited 7 and 8 *(referring to hand shown on currently-displayed screen).* . . .What you could say is what hands would beat that, in general, right?

DT1: Okay, so, like, how about say the rank of the hand? 'Cause there's different ranks like high card, pair, two pair, three pair.

I: Yeah!

Catching on to the idea of in-context help, another member of the design team points out another opportunity:

DT2: In this scenario (points to currently-displayed screen), it's not really wise to fold on the flop 'cause why would you fold on the flop?

I: Ah, there you go. See, there's some strategy that the beginner would need to know. Say, "Hint: It's not really wise to fold on the flop." (refers to area of currently-displayed screen)

Inspired by this suggestion to display in-context hints, design team members start to unpack the strategies for when to "fold" and when to "check." Once these strategies are revealed, another audience member sees an opportunity to build these strategies right into the interface through an interface constraint:

A3: Make it a dynamic button that says “check” when check is appropriate, and change it to “fold” when fold is appropriate.

DT1: Okay.

Concerned that it should still be possible to apply an inappropriate strategy while learning the game, participants contemplate other ways the interface might communicate strategies and tips while the game is being played. This culminates in the following exchange:

A4: I think this would be good. . .You ever watch the music videos where they have the little pop-ups? You know, this happened during the filming or whatever. That’s kind of the help system. So as [something happens], a little pop-up bubble [appears]: “This is this.” So when you hover over the check/fold button, [you could show] another bubble.

I: Yeah, explain the expert’s behavior at the end. You could say. . . why the expert chooses to fold. So try to get into the head of the expert.

DT1: Pot odds or something? Because pot odds are, like, what you have to risk based on what is in the pot.. . .An expert knows what that is, but a beginner wouldn’t.

6. DISCUSSION

We now turn to a discussion of the findings vis-à-vis the four research questions posed by this research. We conclude this section with a general discussion of our findings.

6.1 RQ1: Do PWs Generate Educationally-relevant Discussions?

The educational benefits of PWs rest in their ability to stimulate discussions in which students’ user interface designs are used as a basis for exploring the design principles and concepts taught in the course. One indication of whether PWs succeeded in this regard can be found in the results of the content analysis: On average, 31% of all PW talk was dedicated to DESIGN TALK. Given that DESIGN TALK was rooted in the user interfaces under review, we see that USER INTERFACE TALK, which constituted 26% of overall talk on average, necessarily served a key complementary role in such discussions. Taken together, DESIGN TALK and USER INTERFACE TALK composed a majority (53%) of all talk—a strong preliminary indication that PW discussions were educationally-relevant.

Delving deeper into the results, we find that 11% of DESIGN TALK (3% of all talk) actually enlisted the design concepts and principles explored in the course. Given that such concepts and principles were a key emphasis of the course within which the PWs were situated, one might be concerned that such a small percentage of DESIGN TALK was actually dedicated to them. However, the PW thrusts participants into a situation of *design practice*, where decision-making is *not* based chiefly on research-based theory [Schön 1990]. Indeed, as Schön [1990], aptly notes, design practitioners engage in a reflective conversation with their design materials, drawing extensively on their personal “repertoire[s] of themes and examples” (pp. 78-79) to make progress. That the design discussions we observed in the PWs contained a mix of theory, practical considerations, and common sense may not only reflect the realities of design practice, but also the authenticity of the design situations considered in the PWs.

6.2 RQ2: To what degree do Students Participate?

The results indicate that the extent to which participants contributed Design Talk varied by speaker type. The instructor contributed the most DESIGN TALK of any other speaker type (44%), followed by student design team members (26%), the test user (17%), and audience members (13%). We believe that this finding, when viewed through the lens of *situated learning theory* [Lave and Wenger 1991], indicates that the PW facilitated opportunities for students to participate in increasingly central ways in design discussions as they took on different roles. At the periphery of the PW activity were student audience members, who contributed the least to the discussions. In this role, students mainly observed the activity. However, when they did contribute, their contributions were most likely to be on the topics that were most relevant to the course: DESIGN TALK and USER INTERFACE TALK. We speculate that, in their roles as somewhat detached observers, audience members were in a good position to focus and reflect on user interface and design issues, without being distracted by the procedural details of the activity.

More centrally involved in the PW activity were the test users, who completed tasks with the interface. One of the key skills to be developed in an HCI course, especially one that consists mainly of computer scientists (as was the case in the courses we studied), is the ability to step away from one's interest in technology development, and into the shoes of users of the technology [Norman 2002]. The PW activity provided students with valuable opportunities to do just that. Test users were actively involved in the activity, contributing about one quarter of the overall talk. Owing to the nature of the role, most of test users' contributions were TASK DESCRIPTION and TASK EXECUTION segments, although they also contributed modestly to DESIGN TALK and USER INTERFACE TALK.

Most centrally involved in the PW activity were design team members, who were charged with describing tasks, simulating their user interface, and ultimately explaining and defending their designs. In this role, students had valuable opportunities to engage in two authentic practices of the software industry. First, they got a taste of what it might be like to run a low fidelity prototype test—an important early evaluation activity. Second, they got a taste of what it might be like to present a preliminary design to a software team with an especially critical eye. Because they were responsible both for describing the tasks to be performed, and for simulating their interface for those tasks, design team members contributed more TASK DESCRIPTION TALK and USER INTERFACE TALK than any other participant role. They also contributed the most PROJECT TALK, which focused on their overall interface design projects, including its background and history.

6.3 RQ3: How, and To What Degree, is Design Justified?

We found that participants attempted to justify 73% of design critiques and suggestions; just 17 percent were unjustified. Some 46% of participants' design critiques and suggestions were supported by *direct* justifications consisting of one or more statements in close proximity to the critiques or suggestions; roughly one-quarter of design critiques and suggestions could be linked to *indirect* or *implicit* justifications that either were not uttered in close proximity to the original design statements, or were uttered by someone other than the person who made the design statements. In addition, we found that 30% of participants' justifications were rooted either in empirical evidence or the design concepts and principles learned in the course; most of the remaining 70%

were grounded in common sense, practical concerns, and intuitions about users. Finally, we could detect no statistically significant differences with respect to the strength and types of the justifications provided by the instructor and students.

We believe that these results furnish a rich, multi-layered response to RQ3. The PWs appeared to promote a culture that placed a high value on justifying design statements. Moreover, at least from a statistical standpoint, learners' justifications were indistinguishable from those of the instructor. Thus, it appears that through the PW activity, learners were able to develop an ability to make linguistic causal arguments regarding design.

These results also raise a broader question: To what extent do the conversations we observed—with their emphasis on linguistic, causal justifications of design—reflect those found in studio-based schools of design practice? As Schön [1990] notes, in the design studio, students frequently struggle with their instructors' critiques based on tacit criteria they have not yet internalized. As one student in his study stated, "It's amazing—intuitively you look at [a design] and you know it's wrong, but it's very hard to get down to the reason" (p. 55). In this case, the student wanted a design rationale, but the instructor was not able to provide one. Reflecting on this student's intuitive understanding of her design situation, the instructor instead constructed "a new problem not from research-based theory, but from his repertoire of themes and examples" (pp. 78-79).

Similarly, as shown in our qualitative analysis of PW design discussions, while most design critiques and suggestions were justified, such justifications served more to motivate the need for re-design than to dictate concrete solutions. This was evident both in the tentative language in which the instructor phrased his critiques and suggestions (e.g., "You might try this"), and in the instructor's tendency to use critiques to raise questions (e.g., "I guess the question is, do people need [scene properties] visible all the time, or only when they are setting them?"). If there was an observable difference between Schön's description of the design studio and our observations of PWs, it lay in the PW instructor's tendency to *raise questions*, rather than *construct new design problems* to be solved.

6.4 RQ4: How and Why Do Design Discussions Vary by PW Session?

In conducting this study, we wondered whether the amount of design discussion would increase in later PW sessions, as participants became more experienced and comfortable with the activity. We found that the distribution of talk across content categories varied significantly by session; however, we found no evidence that DESIGN TALK increased from session to session. Instead, we identified a significant positive correlation between session length and the amount of DESIGN TALK: Longer-running sessions tended to be accompanied by increased amounts of DESIGN TALK.

Our qualitative analysis identified three features that influenced the amount of design talk in a given PW: (a) the familiarity of the design domain; (b) the clarity of the design domain, tasks, and interface; and (c) the extent to which test users actually encountered user interface problems. In general, more familiar design domains, clearer tasks and user interfaces, and the presence of test user struggles all led to increased design discussions. Those discussions revolved around three distinct themes: (a) how to make functionality more accessible; (b) what functionality is most important; and (c) how to support

learning. All of these are central themes in courses that focus on user interface design. That they were central to PW design discussions provides further evidence that the PW activity considered in this study succeeded in stimulating educationally-relevant conversations.

6.5 General Discussion

The study presented in this article raises at least four key issues that warrant further consideration: *researcher bias, generalizability, evidence of student learning,* and *lack of comparison data.* In the general discussion that follows, we address these issues in turn.

6.5.1 Researcher Bias. The first issue raised by this study relates to the fact that the course instructor in the study is also the first author of this article. He arrived at the idea of the prototype walkthrough activity, as described in this paper, through several years of trying and refining the activity in his HCI courses. He decided to videotape the sessions analyzed here as a way of exploring the activity in greater depth, so that it could be improved in future offerings of the course. In this sense, this study began as a form of *action research* [Elliott 1991], without a concrete research agenda other than to improve the first author's teaching of the course. The research questions and coding scheme that form the foundation of the study emerged over a year after the last session was videotaped, through a collaborative partnership with the third author—a co-researcher without a vested interest in the course..

In reflecting on this sequence of events, we can make two statements about the potential bias introduced by the first author being a participant in this research. First, his knowledge of this research could not have influenced his behavior in the prototype walkthrough sessions studied, since he had no knowledge of the research a priori. Second, his involvement as a participant in this study may have colored the analysis of the study sessions, especially with respect to the qualitative analysis.

While this potential bias must certainly be taken into consideration when interpreting the study's results, we have taken two steps to mitigate it. First, the study's research questions and original coding scheme were developed in partnership with a disinterested third party (the third author) based upon collaboratively viewing segments of the video record. Second, we have performed a rigorous inter-rater reliability analysis to verify the reliability of our coding scheme. In that analysis, the second author, who was not a participant in the research (and who, in fact, is not even a computer scientist) achieved a high level of agreement with the first author. This indicates that, even if the coding scheme was somehow biased by the first author's involvement as a participant in the research, it is unlikely that such a bias influenced the quantitative analyses of the PW sessions.

6.5.2 Generalizability of Study Findings. A second key issue raised by our study relates to the extent to which its findings generalize. Because the study considered two offerings of an HCI course taught by the same instructor and the same university, one clearly needs to be cautious in generalizing the study's findings beyond the courses studied. At the same time, three features of the study suggest that its results should be applicable beyond the specific courses studied, as discussed below.

Study focused on common and standard course. The context of this study was an upper-division course on human-computer interaction. According to the most recent survey data available [McCauley and Manaris 2002], 40 percent of accredited undergraduate degree programs in the U.S. offer an upper-division HCI course. In its

most recent "Curricula for Human-Computer Interaction " [Hewett et al. 1996], the Association for Computing Machinery's Special Interest Group on Computer-Human interaction (ACM SIGCHI) establishes curricular standards for such a course. Inspection of those standards suggests that the HCI course that provided a backdrop for our study aligns well with the "CS 1" HCI course described in the standards. Thus, there is good reason to believe that the course is representative of HCI courses taught at other U.S. undergraduate institutions.

Prototype Walkthrough Focuses on Core HCI Skills. While the PW activity, as proposed here, may not be widely used in current HCI educational practice, we believe it has broad applicability in HCI education. An analysis of the activity vis-à-vis the core topics included in ACM SIGCHI's "CS 1" curricular recommendations suggests that the PW activity is directly relevant to 5 out of the 10 core topics—most importantly, "Interface quality and evaluation" and "Interface design project: presentations and discussion." Given the primacy of these topics in standard HCI curricula, it follows that the PW activity should have broad appeal among HCI educators.

Study results triangulate with other studies of design crits in HCI education. The study presented here constitutes the most ambitious attempt to study the moment-to-moment interactions of design crits within HCI education. However, as discussed in Section 2, we are by no means the first to recognize the appropriateness of design crits for HCI education. As compared to the (less detailed) descriptions of design crits reported in the literature [Reimer and Douglas 2003; Cennamo et al. 2011; Reimer et al. 2012; Greenberg 2009], the findings reported here appear to be consistent in at least three key respects. First, previous studies have emphasized the rich opportunities that design crits afford for considering multiple design perspectives. In reflecting on their studies of design crits, Cenamo et al. [2011] were emphatic about this:

> In the HCI classes we observed, one very effective technique for evaluating designs involved students putting themselves into the role of the user in order to switch perspectives and experience how their own designs might be interpreted and experienced from different viewpoints. Students can use this technique during the initial generation of design ideas as well; carefully considering the design problem from multiple perspectives can effectively open up new possibilities for productive solutions. (p. 654)

Second, past work has emphasized the value of design crits in *modeling* the process of critically reviewing a design. This was a key theme in the study of Reimer et al [2012]. As one participant in their study stated,

> I think the in-class [design crits] helped to clarify exactly what was expected of us. In particular, [they] helped me think about how to generate new ideas. For each new design, I tried to think of reasons why that particular design was both good and bad, and then tried to come up with ways to improve the design. (p. 628)

Finally, past studies have underscored the value of design crits in requiring students of computer science to shift their focus from implementation to design. Both Reimer et al. [2012] and Greenberg [2009] observed that computer science students tend to be preoccupied with implementation technologies, and argue that this is to their detriment when it comes to developing design skills. For this reason, both found the process of

having students construct and critically review low fidelity prototypes made out of simple media to be educationally valuable. As Greenberg [2009] noted, "As a consequence of working with [low fidelity materials], students could spend their time thinking about design rather than programming, and could rapidly iterate over designs…as others critiqued early versions" (p. 31).

In sum, the more general findings of related studies of design crits in HCI education lend credence to our study's results, which can be seen to provide a more rigorous account of the ways in which one form of design crit (the PW) actually facilitates the learning of design and evaluation skills within the context of an HCI course. Through both our quantitative and qualitative results, this study furnishes a rich account of the learning processes that promote the development of such skills.

6.5.3 Evidence of Student Learning. A third key issue raised by our study relates to the question of whether students actually learned through the PW activity. Ideally, in addition to its detailed analysis of student learning ***processes***, our study would have collected quantitative measures of student learning ***outcomes***—for example, pre to post improvement on a test of design knowledge or an assessment of design quality. However, because of the exploratory nature of this research, we did not include such a rigorous assessment of design learning as a dependent variable in the study. Such an assessment would have strengthened this study, and should be a key priority for future research.

Nonetheless, through the lens of Situated Learning Theory [Lave and Wenger 1991], our detailed analysis of student learning processes furnishes an alternative form of evidence of student learning. From this perspective, learning is seen in terms of increasingly central participation in community practices. As our analysis showed, students participated robustly in the PW activity by presenting, justifying, and critiquing user interface design in varied and increasingly central roles: as observers, discussants, test users, and presenters (see esp. Section 6.2). According to Situated Learning Theory, this analysis of participation, in itself, furnishes evidence of learning.

6.5.4 Lack of Comparison Data. A fourth key issue raised by our study relates to the lack of baseline data against which to interpret our detailed analyses of PW discussions. Indeed, in the absence of a similar analysis of comparable design discussions, it is difficult to draw definitive conclusions regarding the efficacy of the PW activity. For example, based on our analysis, an HCI educator may wonder whether to be concerned about the finding that just 3% of all design justifications were grounded in principles, theory, or empirical evidence. Is this in line with the way in which design is justified in other settings?

We believe our study results would be more meaningful if they could be compared against another design discussion setting, such as naturally occurring talk in an HCI course, or design meetings in an industrial setting. Unfortunately, we are unaware of detailed content analyses of discussions in such settings. Thus, we are left either to collect and analyze the data ourselves, or to report on what we have, in the hope that it can provide a foundation for future research in the field. In the interest of time and moving this line of inquiry forward, we have chosen the latter approach, while emphasizing that future research on design discussions in comparable settings is clearly needed.

7. IMPLICATIONS AND FUTURE WORK

This study's findings have implications for both HCI education practice and research. With respect to practice, the findings furnish at least three key pieces of empirical

evidence that might motivate HCI educators who are considering the use of the PW in their own courses:

- PWs promote pedagogically-relevant discussions and active student participation.
- PWs provide students with opportunities to apply their emerging HCI design knowledge by grounding their design statements in empirical evidence and established theories and principles.
- PWs provide opportunities for *increasingly central participation* [Lave and Wenger 1991] in authentic practices that rival those they may ultimately encounter in the profession.

In addition, the findings provide HCI educators with guidance on *how* they might actually tailor the PW for maximum effectiveness in their own courses. This includes not only a general protocol for running the activity, but also the following best practices for implementing it:

- *Use familiar design domains.* Encourage student design teams to focus on design domains that are familiar and personally meaningful to those in the class. More familiar and personally meaningful design domains will increase the likelihood that other students take interest and participate in the design crits of their peers. For example, in our study, while the design of a DVR remote control generated extensive discussion, the design of a power utility mapping tool for a local power company did not.
- *Require clear interface and task descriptions.* Emphasize the need for students to present their design domain, user interface, and tasks in clear and understandable terms. This will increase the chances that design crits maintain a focus on actual design issues, rather than devolving into discussions that clarify what is being designed for, what tasks are to be performed, and what the user interface actually does, as was the case in some of the PWs we studied. To ensure that students interfaces and tasks are clearly described and presented, consider performing a preliminary review of the materials before the PWs take place.
- *Leverage test user struggles as opportunities for learning.* The PW sets up a situation in which a test user performs tasks with a prototype user interface in a public forum. From the standpoint of HCI education, the best thing that could occur in this situation is for the test user to struggle, as such struggles provide invaluable opportunities for students and instructors to engage in grounded design discussions that enlist the principles, theories, and concepts being explored in the course. Embrace these opportunities by initiating design discussions that consider those struggles in detail. As this study found, a viable strategy entails (a) eliciting the test user's perspective to provide further evidence of the struggles, (b) determining the cause of those struggles by tying them to specific design features, and finally (c) identifying course principles, theories, and concepts that might be enlisted both to explain the struggles and motivate alternative designs.

Finally, with respect to HCI education research, this study follows a strong tradition in the learning sciences of interpreting the moment-to-moment interactions of classroom activities both through the lens of existing learning theory [e.g., Sawyer and Berson 2004], and for the purpose of constructing new theory [e.g., Suthers and Hundhausen

2003]. It constitutes one of the few attempts, within HCI education research, to apply rigorous video analysis techniques in order to better understand, and ultimately to improve, HCI education practice. In so doing, the study demonstrates the potential for video analysis to make computing education research in general, and HCI education research in particular, more rigorous and theoretically-grounded.

This study motivates several directions for future research. We highlight three of these here. First, while this study carefully considered design discussions, it did not track the extent to which project teams actually made design changes based on these discussions, nor did it collect interview or survey data on participants' attitudes and experiences. In future work, we believe it will be important to collect these two sources of data. Collecting data on design changes will allow us to explore systematically the extent to which students actually integrate the design knowledge on which PW design discussions focus. Likewise, collecting interview and survey data will allow us to triangulate the results of our video analysis with reports of students' subjective experiences. We are particularly interested in collecting pre/post attitudinal data on students' *self efficacy* with respect to user interface design. The PW activity provides students with opportunities to assess themselves relative to others. According to *self- efficacy theory* [Bandura 1997], such opportunities are crucial both to forming accurate perceptions of their own abilities. They also increase the chances that students will persist in the discipline [Rosson et al. 2011]

Second, as discussed in Section 6.5.3, in order to provide a basis for comparing our results, future research will need to perform similar detailed analyses of design discussions in alternative settings. One obvious setting to consider is HCI education. Here, the key question relates to how best to foster educationally beneficial design discussions. To provide baseline for comparison, one could study naturally occurring design talk in traditional lecture settings. One could then systematically vary features of the PW activity, in order to determine their impact on design talk. For example, in the design crits studied by Reimer and Douglas [2003], a test user was not explicitly involved. Does the presence of a test user who actually interacts with students' prototypes significantly enhance design talk? Because HCI education endeavors to train students become industry professionals, another obvious setting to consider is industry. How do industrial design teams talk about design? To what degree are design decisions based on empirical evidence and design principles, and to what degree are they based on other factors?

Finally, given that the PW activity requires more classroom time than many HCI instructors may have available, we are intrigued by the possibility of conducting PWs online. One approach would be for design teams to post video recordings of a walkthrough with a test user. PW discussions could then be conducted through an online environment that supported asynchronous discussions anchored to specific points in the video. To support this kind of asynchronous interaction, we have developed OSBLE (see http://osble.org), an online learning management environment designed specifically to support SBL activities such as the PW. In future research, we would like to perform an empirical comparison of asynchronous and synchronous PWs, in order to determine whether the benefits of PWs observed in this study can be harnessed on a broader scale in asynchronous learning environments.

ACKNOWLEDGMENTS

This research is funded by the National Science Foundation under grant nos. CNS-0721927 and CNS-0939017. Contributions to this work by other members of the research team—Michael Trevisan, N. Hari Narayanan, Dean Hendrix, Martha Crosby, Margaret Ross, and Rita Vick—are gratefully acknowledged. Collaboration between C. Hundhausen and M. Petre was supported by M. Petre's Royal Society Wolfson Research Merit Award. We are grateful to the anonymous reviewers of this article and the conference paper on which it is based [Hundhausen et al. 2011] for many helpful suggestions that we were able to incorporate into this article—especially the comparison of our results and Schön's [1990] observations of the design studio.

REFERENCES

ARVOLA, M. AND ARTMAN, H. 2008. Studio life: The construction of digital design competence. *Digital Kompetanse 3*, February, 78–96.

BANDURA, A. 1997. *Self-efficacy: the exercise of control*. Worth Publishers.

BARKER, P. 2011. Soft skills important for IT job candidates. *Montreal Gazette*. http://www2.canada.com/montrealgazette/news/archives/story.html?id=a90c9d79-48ac-4890-b505-89fd4c0cc706.

CARD, S.K., MORAN, T.P., AND NEWELL, A. 1983. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, Hillsdale, NJ.

CENNAMO, K., DOUGLAS,, S., VERNON, M., ET AL. 2011. Promoting creativity in the computer science design studio. In: *Proc. 42nd ACM Technical Symposium on Computer Science Education*. ACM, New York, 649–654.

CROSS, N. 2001. Design cognition: results from protocol and other empirical studies of design activity. In: C.M. Eastman, W.M. McCracken and W.C. Newstetter, eds., *Design Knowing and Learning: Cognition in Design Education*. Elsevier Science, Oxford, 79 – 103.

DOCHERTY, M., SUTTON, P., BRERETON, M., AND KAPLAN, S. 2001. An innovative design and studio-based CS degree. In: *Proc.32nd SIGCSE technical symposium on computer science education*. ACM, New York, 233–237.

ELLIOTT, J. 1991. *Action research for educational change*. Open University Press, Philadelphia.

GREENBERG, S. 2009. Embedding A Design Studio Course in A Conventional Computer Science Program. In: P. Kotze, W. Wong,, J. Jorge, A. Dix and S.P. Alexandra, eds., *Creativity and HCI: From Experience to Design in Education*. Springer, 23 – 41.

GUZDIAL, M. 1994. Software-realized scaffolding to facilitate programming for science learning. *Interactive learning Environments 4*, 1, 1–44.

HEWETT, T., BAECKER, R., CARD, S., ET AL. 1996. *ACM SIGCHI Curricula for Human-Computer Interaction*. ACM, New York.

HEWSON, R. 1994. Marking and making: A characterisation of sketching for typographic design. Unpublished Ph.D. Dissertation, The Open University, Milton Keynes, U.K.

HUNDHAUSEN, C.D., BALKAR, A., NUUR, M., AND TRENT, S. 2007. WOZ Pro: a pen-based low fidelity prototyping environment to support wizard-of-oz studies. In: *Extended Abstracts: 2007 ACM Conference on Human Factors in Computing Systems*. ACM press, New York, 2453–2458.

HUNDHAUSEN, C.D., FAIRBROTHER, D., AND PETRE, M. 2009. Studying Prototype Walkthroughs in a Human-Computer Interaction Course: Video Analysis Manual (ver. 20). http://publicationslist.org/data/helplab/ref-90/PW-VideoAnalysis-Manual-v20.pdf.

HUNDHAUSEN, C.D., FAIRBROTHER, D., AND PETRE, M. 2011. The "prototype walkthrough": a studio-based learning activity for human-computer interaction courses. In: *Proceedings*

2011 ACM International Computing Education Research Workshop. ACM Press, New York, 117–124.

HUNDHAUSEN, C.D., NARAYANAN, N.H., AND CROSBY, M.E. 2008a. Exploring studio-based instructional models for computing education. In: *Proc. 39th SIGCSE Technical Symposium on Computer Science Education*. ACM Press, New York, 392–396.

HUNDHAUSEN, C.D., TRENT, S., BALKAR, A., AND NUUR, M. 2008b. The design and experimental evaluation of a tool to support the construction and wizard-of-oz testing of low fidelity prototypes. In: *Proc. 2008 IEEE Symposium on Visual Languages and Human-Centric Computing*. IEEE, Piscataway, NJ, 86–90.

KEHOE, C.M. 2001. Supporting critical design dialog. Unpublished Ph.D. Dissertation, Georgia Institute of Technology, Atlanta.

KRIPPENDORFF, K. 1980. *Content analysis: an introduction to its methodology*. Sage Publications, Beverly Hills.

LAVE, J. 1993. The practice of learning. In: *Understanding Practice: Perspectives on Activity and Context*. Cambridge University Press, Cambridge, 3–32.

LAVE, J. AND WENGER, E. 1991. *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press, New York.

LIDWELL, W., HOLDEN, K., AND BUTLER, J. 2010. *Universal Principles of Design*. Rockport Publishers, Berverly, MA.

MAULSBY, D., GREENBERG, S., AND MANDER, R. 1993. Prototyping an intelligent agent through Wizard of Oz. In: *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*. ACM, New York, 277–284.

MCCAULEY, R.A. AND MANARIS, B. 2002. *Comprehensive Report of the 2001 Suvey of Departments Offering CAC-Accredited Degree Programs*. Department of Computer Science, College of Charleston, Charleston, SC.

MILLER, G.A. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review 63*, 2, 81–97.

MYNENI, L., ROSS, M., HENDRIX, D., AND NARAYANAN, N.H. 2008. Studio-based learning in CS2: An experience report. In: *Proc. 46th ACM southeast conference (ACM-SE 2008)*. ACM Press, New York, 253–255.

NIELSEN, J. 1992. Finding usability problems through heuristic evaluation. In: *Proceedings of ACM CHI'92 Conference on Human Factors in Computing Systems*. 373–380.

NORMAN, D.A. 2002. *The Design of Everyday Things*. Basic Books, New York.

POLSON, P.G., LEWIS, C., RIEMAN, J., AND WHARTON, C. 1992. Cognitive Walkthroughs: A Method for Theory-Based Evaluation of User Interfaces. *International Journal of Man-Machine Studies 36*, 5, 741–773.

PREECE, J., ROGERS, Y., AND SHARP, H. 2002. *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, New York.

REIMER, Y.J., CENNAMO, K., AND DOUGLAS, S.A. 2012. Emergent themes in a UI design hybrid-studio course. *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, ACM, 625–630.

REIMER, Y.J. AND DOUGLAS, S.A. 2003. Teaching HCI design with the studio approach. *Computer Science Education 13*, 3, 191–205.

ROSCHELLE, J. 1996. Designing for cognitive communication: Epistemic fidelity or mediating collaborative inquiry? In: D.. Day and D.K. Kovacs, eds., *Computers, Communication and Mental Models*. Taylor & Francis, London, 13–25.

ROSSON, M.B., CARROLL, J.M., AND SINHA, H. 2011. Orientation of Undergraduates Toward Careers in the Computer and Information Sciences: Gender, Self-Efficacy and Social Support. *Trans. Comput. Educ. 11*, 3, 1–23.

SACHS, A. 1999. "Stuckness" in the design studio. *Design Studies 20*, 2, 195–209.

SAWYER, R.K. AND BERSON, S. 2004. Study group discourse: How external representations affect collaborative conversation. *Linguistics and Education 15*, 387–412.

SCHÖN, D. 1990. *Educating the reflective practitioner*. Jossey-Bass Publishers, San Francisco.

SCHUMANN, J., STROTHOTTE, T., RAAB, A., AND LASER, S. 1996. Assessing the effect of non-photorealistic rendered images in CAD. In: *Human Factors in Computing Systems: CHI 96 Conference Proceedings*. ACM Press, New York, 35–41.

SHNEIDERMAN, B. 1983. Direct manipulation: a step beyond programming languages. *IEEE Computer 16*, 8, 57–69.

SUTHERS, D. AND HUNDHAUSEN, C. 2003. An experimental study of the effects of representational guidance on collaborative learning processes. *Journal of the Learning Sciences 12*, 2, 183–219.

TOMAYKO, J.E. 1996. Carnegie Mellon's software development studio: a five year retrospective. In: *Proc. 9th Conf. on Software Engineering Education*. IEEE Computer Society, Los Alamitos, CA, 119.

WILSON, J. AND ROSENBERG, D. 1988. Rapid prototyping for user interface design. In: M. Helander, ed., *Handbook of Human-Computer Interaction*. North-Holland, New York, 859–875.

Differences between this article and previously published work

This article expands upon a conference paper entitled "The Prototype Walkthrough: A Studio-Based Activity for Human-Computer Interaction Courses" that was published in the *Proceedings of the 2011 ACM International Computing Education Research Conference*. While excerpting from key sections of that conference paper (including the abstract, introduction, related work, results, and discussion), this article contains significant new material:

- We have expanded the results section (Section 4) by adding two new subsections:
  - Section 4.3, which presents a new analysis of justification strength.
  - Section 4.4, which presents a new quantitative and qualitative analysis of session-by-session differences. (This is the most significant expansion of the original conference paper.)
- We have added a new qualitative analysis of the session features that influenced design talk, and of the themes of design discussions (Section 5).
- We have expanded the Discussion section (Section 6) to incorporate a deeper theoretical analysis of our results.
- We have added an explicit discussion of our study's implications for human-computer interaction education research and practice, and expanded our discussion of future work (Section 7).

In sum, this article can be seen as a greatly-expanded version of our original ICER 2011 conference paper. We conservatively estimate that the article contains 30 percent new material, thus providing an expanded archival-quality counterpart to our ICER 2011 paper.